Real-Time Intrusion Detection and Tracking in Indoor Environment Through Distributed RSSI Processing

Ossi Kaltiokallio and Maurizio Bocca
Aalto University School of Electrical Engineering
Department of Automation and Systems Technology
Helsinki, Finland
{ossi.kaltiokallio, mauzio.bocca}@tkk.fi

***Abstract*—In the context of wireless sensor networks, the received signal strength indicator has been traditionally exploited for localization, distance estimation, and link quality assessment. Recent research has shown that, in indoor environments where nodes have been deployed, variations of the signal strength can be exploited to detect movements of persons. Moreover, the time histories of the received signal strength indicator of multiple links allow reconstructing the paths followed by the persons inside the monitored area. This approach, though effective, requires the transmission of multiple, raw received signal strength indicator time histories to a central sink node for off-line analysis. This consistently increases the latency and power consumption of the system. This work aims at applying distributed processing of the received signal strength indicator for indoor surveillance purposes. Through distributed processing, the nodes are able to autonomously detect and localize moving persons. The latency and power consumption of the system are minimized by transmitting to the sink node only the alerts related to significant events. Moreover, power consumption is further reduced through a high accuracy time synchronization protocol, which allows the nodes to keep the radio off for 60 % of the time. During the tests, the system was able to detect the intrusion of a person walking inside the monitored area and to correctly track his movements in real-time with a 0.22 m average error. Possible applications of this application include surveillance of critical areas in buildings, enhancement of workers safety in factories, support to emergency workers or police forces in locating people e.g. during fires, hostage situations or terrorist attacks.**

***Keywords*—RSSI; intrusion detection; tracking; distributed processing; real-time monitoring**

I. INTRODUCTION

Despite becoming the enduring technology for navigation and tracking purposes, GPS systems present also some limitations, e.g. a null or very limited functionality in indoor environments and the need of having the users carrying a device at all times. For these reasons, device-free localization (DFL) of people in indoor environments has recently attracted the attention of the research community. DFL could be exploited in a variety of applications, e.g. tracking of customers in shopping malls with the aim to analyze their reaction to products and advertisements placement, detection of intruders in critical buildings, localization and tracking of besiegers and hostages during sieges and of civilians during fires, and for workers safety enhancement in industrial halls with moving machineries. Wireless sensor networks (WSNs) provide the means to perform these tasks by extracting useful information from the variations of the received signal strength indicator (RSSI) caused by the presence and movements of people inside the monitored area. Since additional sensors such as cameras or infrareds are not used in this type of systems, the nodes composing the network can be considered as radio frequency (RF) sensors [1]. Compared to wired surveillance systems, WSNs consistently reduce both the installation time and cost, and can be easily redeployed in another area of interest if needed. Moreover, the limited cost of a unit makes it possible to deploy a large number of nodes to possibly cover extensive areas.

In the past, the RSSI has been exploited for nodes localization [2], [3], distance estimation [4], [5], and link quality assessment purposes [6], [7]. However, it has also proved to be useful for detecting the movements of individuals found inside the monitored area [8], [9], [10]. In stable conditions, i.e. when no person is found inside the network, the RSSI measurements are nearly constant [7]. On the contrary, the presence and movements of people found inside the network introduce variance in the RSSI measurements. By monitoring the changes of the RSSI over multiple intersecting links, it is then possible to create a virtual RF mesh through which to detect and track individuals.

This paper describes a WSN for real-time intrusion detection and tracking by distributed processing of the RSSI signals. Differently than in other previous works in which the nodes of the network transmit all the collected RSSI measurements to the sink node, in the proposed system a distributed algorithm enables the nodes to transmit only those alerts related to significant events. The data received at the sink node are than combined and processed in real-time to produce accurate estimates of the current position of the intruder. A Kalman filter is applied to improve the tracking accuracy and smoothness. Furthermore, other previous works rely on collecting measurements and training the system in static conditions (i.e. monitored area is vacant) before deployment, thus making it impossible to

be used in emergency response scenarios. The approach presented in this paper doesn't rely on training the system and is ready to be used once deployed.

The distributed algorithm reduces the amount of packets the nodes have to transmit to the sink node, and consequently the nodes power consumption, extending the overall lifetime of the system. Power consumption is further reduced by means of a high accuracy time synchronization protocol, enabling TDMA communication among the nodes. Being time synchronized, the nodes can activate their radio only in correspondence with the scheduled transmissions.

The remainder of this section presents the related work, describes the key challenges to be faced in the development of such a system, and finally lists the contributions of the paper. Section II covers the wireless side of the DFL system, i.e. the applied time synchronization protocol, the radio management scheme and the distributed RSSI processing algorithm. Section III introduces the ellipsoidal alert model used to localize the person and the Kalman filter that is used to enhance tracking accuracy. The performance and accuracy of the system are presented and discussed in Section IV. Future work directions are announced in Section V, and conclusions are drawn in Section VI.

A. Related Work

Several studies have demonstrated that the RSSI can be exploited to detect and track an object found inside the area monitored by a WSN. Hussain et al. [11] exploit the RSSI to detect an intruder positioned between two communicating nodes. In their work they show a high level of correlation between the alerts raised by a RSSI-based intrusion detection algorithm and the crossings of the line-of-sight (LoS) radio link. The results were validated using a beam of light traveling from one node to the other and a high level of correlation between the light beam crossings and the RSSI-based algorithm was obtained. In [12], it is shown how a moving person causes multipath fading and shadowing of the radio signal.

By monitoring the changes in the attenuation field of the network, Wilson and Patwari are able to estimate the position of individuals found inside an area surrounded by nodes [8]. In their work, the system is first calibrated in static conditions, i.e. when no object is found in the monitored area. At run-time, the results of the calibration phase are used to derive the differences in the RSSI measurements, and Tikhonov regularization is applied to obtain the position estimates.

In [9], the statistical attenuation model and the variance of the attenuation field are exploited to obtain the position estimate of the intruder. In addition, a Kalman filter is applied off-line to data previously collected for tracking the movements of the person. In the experiments the nodes are deployed around a house. The average tracking error is 1.03 m. Most of the error is due to the lag introduced by the Kalman filter, but also to delays caused by RSSI measurements collection and processing. Without these delays, the average tracking error is reduced to 0.45 m.

Zhang et al. [10] propose a RF network in which the tracking is based on the difference between the RSSI measurements collected by the nodes during a training phase carried out in static conditions and the measurements collected in a dynamic environment, i.e. when a person is moving inside the monitored area. However, the reported latency of the system in localizing a person in the corresponding position is 3 s. This fact makes it impossible for the system to correctly detect and track an object moving at a relatively high speed. The reported accuracy, in a 4x4 nodes grid, is 0.66 m.

The same approach adopted in [10] is further developed in [13] for a system capable of tracking multiple persons simultaneously moving in the monitored area. In it, the nodes detecting a significant change in their RSSI values autonomously organize into clusters. At first, the cluster head gathers the RSSI measurements collected by the members of its cluster. Then, it computes the probability of detection of an object. Finally, it transmits this value back to the sink node. In this way the amount of data transmitted to the sink node is reduced. However, due to the cluster formation and inter-cluster communication overhead, the reported latency of the system is 2 s. The reported accuracy is 0.85 m in the one person case and 1.08 m in the two person case.

In the above-mentioned studies, the capability to detect and track an object relies on a previous calibration of the system, which increases its deployment time and its power consumption [8], [9], [10], [13]. Besides, the long-term usability of such RSSI model-based systems would also suffer from the fact that marginal changes of the conditions of the monitored indoor environment (e.g. furniture placement, doors and windows openings and closing) can dramatically modify the propagation patterns of radio signals and thus the RSSI measurements collected by the nodes, ultimately leading to the need of retraining the model. The system proposed in this paper does not rely on a previous calibration in static conditions, and furthermore it dynamically adapts to the changing conditions of the monitored area.

Also, the power consumption of the nodes, which ultimately defines the lifetime of the system, is not taken into consideration, since each node keeps the radio on at all times. On the contrary, in the proposed system the nodes, which are time synchronized, disable the radio when scheduled communications are not expected. In addition, the transmission of all the raw RSSI measurements [8] [9] collected by the nodes to the sink node for off-line data analysis, though effective for tracking, is not power efficient, since radio transmissions and receptions are the most power-hungry operations performed by the nodes. This approach considerably increases the latency of the system [10] [13], especially in multi-hop scenarios. These facts restrict the applicability and utility of these systems in those emergency situations such as fires, sieges, or terrorist attacks, when real-timeliness becomes a critical requirement.

B. Challenges

There exist three main challenges in the development of a WSN for real-time intrusion detection and tracking based on RSSI processing. In the first place, in typical indoor environments, the transmission range and the area of successful packet delivery of the nodes is limited. In unobstructed areas, the transmission range of IEEE

802.15.4 low-power radios can reach 70 m, whereas in highly obstructed indoor environments, reliable communications can be guaranteed only for short distances, usually in the range of 10 m [14].

A second challenge is represented by the existing link asymmetries and the unpredictability of the RSSI measurements collected by different nodes. Large and short-time decreases in the RSSI values are caused by the shadowing corresponding to LoS crossings, whereas lasting fluctuations of the RSSI values are due to the multipath fading caused by the movements of a person in the close proximity of the LoS links [1]. If the multipath components of the radio signal are spread spatially, i.e. for nodes pairs located e.g. 10 m apart, the intruder crossing the LoS causes a small variation of the RSSI values, making the detection challenging. This fact sets critical limits to nodes spacing and deployment in such an intrusion detection system.

The time-varying performance of the transmitting and receiving nodes, as well as the orientation of the antennas contribute to the unpredictability of the RSSI measurements [15]. Because of this, the developed embedded intrusion detection algorithm does not rely on pre-existing RSSI models of the environment previously built in static conditions.

Finally, the strict real-time requirements in delivering packets to the base station pose another challenge to the system. Communication delays and outages between the sensing devices and the sink node strongly influence the capability of accurately detecting and tracking intrusions [16], [17].

C. Contributions

This paper, which extends the work presented in [18], introduces a RF sensor network for real-time intrusion detection and tracking through distributed RSSI processing. The contributions can be summarized as follows:

- An algorithm, executed locally in the nodes, processing in real-time the RSSI measurements to detect intrusions of individuals inside the monitored area. The algorithm minimizes the volume of traffic towards the sink node, i.e. only alerts related to significant events are transmitted. Moreover, the algorithm operates iteratively, minimizing memory footprint and power consumption of the nodes (no data storage in an external Flash memory or SRAM is required). Execution time of the algorithm is further minimized by using only integer type variables.
- Through a high accuracy time synchronization protocol (error constantly below ±5 μs), the nodes can optimally disable the radio when transmissions are not scheduled, minimizing their power consumption. The activation and deactivation of the radio is performed at the media access control (MAC) layer to minimize the waste of energy. This fact leads to an 80 % increment of the expected lifetime of the system compared to the case in which the radio is always enabled.
- A method to efficiently transmit the information related to intrusion alert notifications to the sink node. The method does not increase the amount of packets transmitted in the network and reduces the latency at the sink node, thus enabling real-time tracking of the intruder.
- A method for estimating the current position of the intruder based on the spatial configuration of the alerts received at the sink node. The tracking accuracy and smoothness is further improved by applying a Kalman filter to the obtained position estimates. The average absolute error in the tracking is approximately 0.22 m also in large obstructed areas.

II. System Architecture

A. Time Synchronization

The wireless intrusion detection system employs the Sensinode U100 Micro.2420 sensor networking platform [19] which is equipped with a low-quality crystal oscillator. The crystal oscillator introduces a significant clock drift (e.g. node A in Fig. 1 drifts 20.3 μs/s) which, in addition, is different in each node (e.g. the relative drift of nodes A and B in Fig. 1 is 8.7 μs/s). The clock drift is also dependent on the nodes age and operating temperature. Therefore, in those applications relying on e.g. an accurately synchronous sampling or radio communications schedule, it becomes essential to precisely synchronize the nodes, not only at the beginning, but also throughout the execution of the application.

The system presented in this work exploits a MAC layer time synchronization protocol, *μ-Sync,* originally proposed in [20]. This protocol, deriving from the flooding time synchronization protocol (FTSP) [21], was initially implemented on a different hardware, i.e. TI CC2431 [22], and it has recently been ported to the Micro.2420 platform [23]. Through this protocol, the nodes are time synchronized with the clock of the sink node, which provides the global time of the network. During the synchronization phase, the sink node broadcasts a series of synchronization beacons at a predetermined rate.

At the reception of the synchronization beacons, the nodes adjust their own clocks to match the global time of the sink node and calculate an estimate of their own clock drift which is then used at run-time in order to guarantee

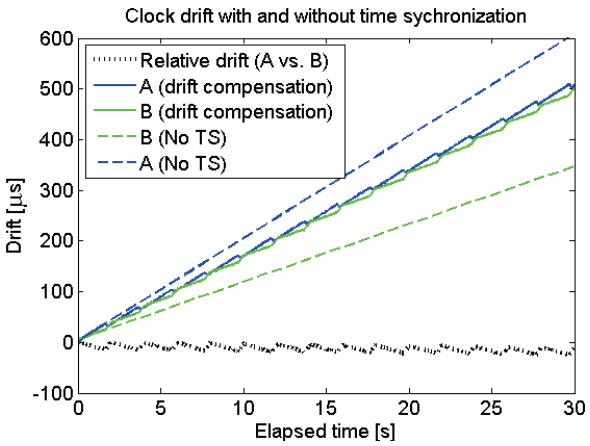


Figure 1. Original (dashed lines) and compensated (solid lines) drift of nodes A and B. Time synchronization allows the nodes to drift at the pace set by the clock of the sink node.

TDMA slotted radio communications. The time synchronization protocol has constantly an error below ±5 μs [23].

B. Radio Management

The accurate time synchronization enables the nodes to turn their radio off when scheduled communications are not expected. The activation and deactivation of the radio is managed at the MAC layer:

- During a transmission slot, the radio is enabled just before the physical transmission of the packet begins. The radio is turned off at the completion of the interrupt service routine (ISR).
- During a reception slot, the radio is enabled at the application layer after a back-off time, equal to the time it takes to prepare and then transmit the packet. The radio is disabled once the packet reception ISR has completed.

During a transmission slot, the radio is enabled on average for 2.30 ms. Correspondingly, during a reception slot, the radio is on for 4.05 ms. The Micro.2420 platform drains approximately 38.4 mA while in transmitting mode and 39.3 mA while in receiving mode and the current consumption drops down to 10.8 mA when the radio is disabled. Thus disabling the radio leads to considerable power savings, increasing the lifetime of the nodes.

The lifetime of the system is compared to the one of a system in which the nodes are not time synchronized, and thus are forced to keep their radio on in order to reliably communicate. In the simulations, the number of nodes composing the network is set to 16, as in the tests carried out in Section IV. The results are shown in Fig. 2.

If the transmission interval of the nodes is set to 10 ms (as in the tests), the lifetime of the time synchronized network is 4.73 days, approximately 80 % more than for the unsynchronized network (2.65 days). By increasing the transmission interval, the lifetime of the synchronized network grows even further in respect to the unsynchronized network (169 % if 30 ms, 200 % if 50 ms), since in the time synchronized network the radio of the nodes can be turned off for longer intervals of time.

C. RSSI ProcessingAlgorithm

When there are no people inside the monitored area, transmitting to the base station the raw RSSI measurements would represent a considerable waste of power. Thus, detecting intrusions by locally processing the RSSI measurements in the nodes and then transmitting to the base station only the alerts related to significant events not only reduces the communication overhead, but also the need of processing measurements which do not hold interesting information.

The proposed embedded intrusion detection algorithm is based on the fact that when motion occurs nearby a wireless link, its multipath components are altered, inevitably increasing the variance in the RSSI measurements. RSSI variance has been previously identified as a promising measure [9] to detect movement in areas monitored by a WSN. In the following paragraphs the embedded RSSI processing algorithm is presented.

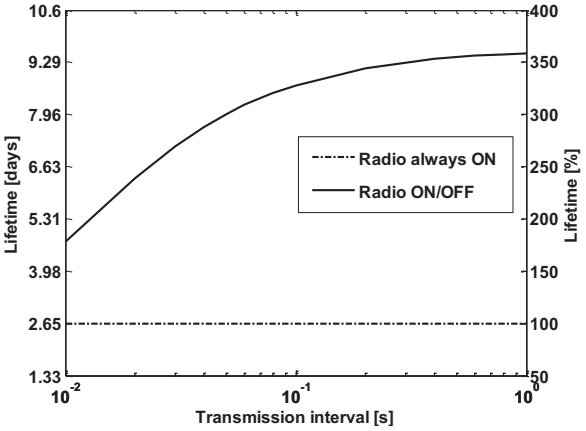


Figure 2. Lifetime of the synchronized and unsynchronized networks.

The change of the n-th RSSI measurement of link i-j is calculated as follows:

$$\Delta RSSI_{i,j}[n] = RSSI_{i,j}[n] - f_{i,j}[n-1], \qquad (1)$$

where $RSSI_{i,j}[n]$ is the current raw RSSI measurement of link i-j (between nodes i and j) and $f_{i,j}[n\text{-}1]$ is the previous filtered RSSI measurement of the same link. The filtered value of sample n is calculated by an exponentially weighted moving average:

$$f_{i,j}[n] = \alpha \cdot RSSI_{i,j}[n] + (1-\alpha)\, f_{i,j}[n-1], \qquad (2)$$

where $\alpha \in [0,1]$ is the smoothing factor (α=0.05 in the tests carried out in Section IV). The sliding sum of the squares of the current and previous measurements is then calculated as follows:

$$\Sigma^2_{i,j}[n] = \Sigma^2_{i,j}[n-1]\left(1 - \frac{1}{n_{forget}}\right) + \Delta RSSI_{i,j}[n]\; RSSI_{i,j}[n] - f_{i,j}[n], \qquad (3)$$

where the forgetting factor n_{forget} defines the size of the window over which (3) is calculated. To calculate the exact value for (3) would require saving entire RSSI time histories over the sliding window, leading to an excessive usage of the RAM memory of the MCU. However, the accuracy of (3) captures variations of the RSSI well within the desired accuracy. The short-term (n_{forget}=5) sliding variance of link i-j at sample n is:

$$\hat{\sigma}^2_{short}[n] = \Sigma^2_{i,j}[n] / (n_{forget} - 1). \qquad (4)$$

The flow of the embedded RSSI processing algorithm is presented in Fig. 3. In it, a second variance $\hat{\sigma}^2_{long}$, is used as a dynamic threshold for triggering the transmission of the alerts and is updated depending on the state of the link.

1. Compute equations (1)-(4);
2. **If** $\hat{\sigma}^2_{short} / 2T \cdot \hat{\sigma}^2_{long} > 1$:
3. **If** $Alert_{i,j} > 0$ & $RSSI_{i,j} > f_{i,j}$:
4. possible LoS crossing over, reset $\Sigma^2_{i,j}$;
5. **Else If** $Alert_{i,j} < 0$ & $RSSI_{i,j} < f_{i,j}$:
6. possible LoS crossing over, reset $\Sigma^2_{i,j}$;
7. **EndIf**
8. $Alert_{i,j} = f_{i,j} - RSSI_{i,j}$;
9. **Else If** $\hat{\sigma}^2_{short} / T \cdot \hat{\sigma}^2_{long} > 1$:
10. $Alert_{i,j} = \hat{\sigma}^2_{short} / \hat{\sigma}^2_{long}$;
11. $\hat{\sigma}^2_{long} = \beta_1 \cdot \hat{\sigma}^2_{short} + 1 - \beta_1 \ \hat{\sigma}^2_{long}$;
12. **Else**:
13. $Alert_{i,j} = 0$;
14. $\hat{\sigma}^2_{long} = \beta_2 \cdot \hat{\sigma}^2_{short} + 1 - \beta_2 \ \hat{\sigma}^2_{long}$;
15. **EndIf**
16. **If** $\hat{\sigma}^2_{long} < 1$:
17. $\hat{\sigma}^2_{long} = 1$;
18. **EndIf**

Figure 3. The flow of the embedded intrusion detection algorithm.

When a link experiences a rapid change (possibly a LoS crossing) in the RSSI measurements, the long-term variance is not updated, since it would decrease the sensitivity of the algorithm over time. In addition, the short-term variance is reset to zero after a possible LoS crossing is over, to assure that the algorithm can capture the next LoS crossing. The branch starting at line 9 is used to capture movement nearby a LoS link or to detect the LoS crossings of links that do not experience a significant change in the RSSI measurements. In this branch the long-term variance is updated (β_1=0.01) so that it adapts slowly to the increase in variance. If the two branches above are not entered, the alert is set to zero and the long-term variance is updated so that it responds to the changes in the RSSI more rapidly (β_2=0.1). Since the algorithm is implemented by using integer variables, $\hat{\sigma}^2_{long}$ is set to one when $\hat{\sigma}^2_{long} < 1$ to assure that variables are not divided by zero and to neglect minor changes in variation.

The Parameter T can be set by the end-user in the network configuration phase. Small values of T will make the algorithm more sensitive (required for large nodes distances and obstructed areas), but will also increase the possibility of false alerts and the need of transmitting alerts to the sink node. On the other hand large values will make the algorithm more robust and minimize the communication overhead, at the cost of losing tracking accuracy. All the experiments described in Section IV have T = 2. In case an alert is triggered, the filtered value $f_{i,j}[n]$ of the link is not updated, since the interest is in the changes of the RSSI in comparison to static conditions (i.e. when no one is in between or in the close proximity of the link). Fig. 4 shows an example of raw and filtered RSSI measurements, short- and long-term variance, and alerts raised by the RSSI processing algorithm.

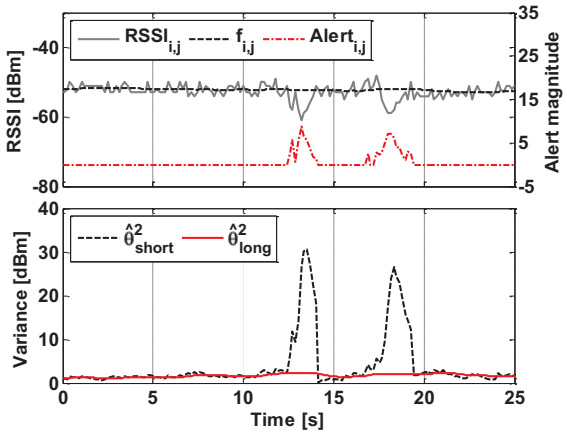


Figure 4. RSSI measurements of a single link experiencing two LoS crossings. The alerts captured by the RSSI processing algorithm are shown above. The short- and long-term variances are shown below.

III. LOCALIZATION AND TRACKING

A. Ellipsoidal alert model

The alert model establishes a relationship between the magnitudes of the alerts, the distance of the person from the LoS link and the a priori known node positions. Wilson and Patwari [9] propose an elliptical model, which assumes that motion occurring inside an ellipse with foci located at the transmitter and receiver will affect the RSSI measurements. Our model makes one simplification to the previous one, i.e. the distance of the node pair does not affect the equatorial radius perpendicular to the LoS.

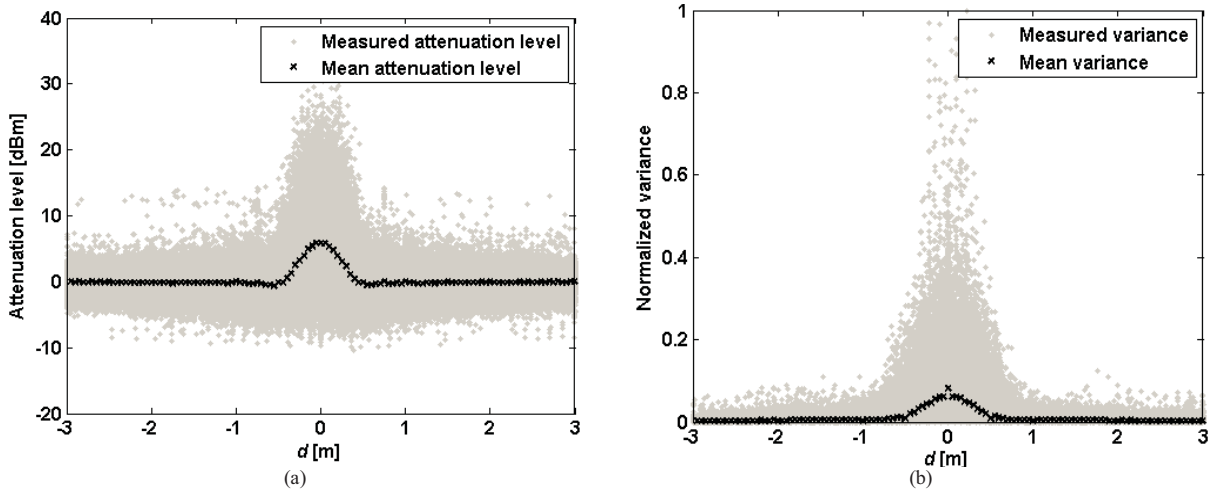
Consider a receiver located at (x_1,0,0) and a transmitter at (x_2,0,0). An ellipsoid connecting the two nodes is expressed as follows:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1, \qquad (5)$$

where a is the equatorial radius between the nodes (a= |x1- x2|), b is the equatorial radius perpendicular to the LoS and c, which is set to unity, is the polar radius along the z-axis.

Our purpose is to determine the exact value of b. To derive it, a broad experiment (663.000 collected measurements) was conducted by placing nodes at various distances, from 1 to 11.3 meters, from a transmitter. Fig. 5a shows the measured and mean attenuation level of the measurements as a function of the person's distance from the LoS connecting the transmitter and receiver. The measured normalized variance and normalized mean variance of the measurements are shown in Fig. 5b.

The mean attenuation level is higher than zero, meaning that the majority of the nodes experience a decrease in the signal strength when the link is crossed, within 0.5 m from the LoS. The nodes measure an increase in the RSSI variance when a person moves within 0.6 m from the LoS. Thus, it is assumed that the distributed algorithm is able



(a) (b)

Figure 5. Measured and mean attenuation level (a), and measured and mean normalized variance (b).

of triggering an alert within these boundaries, therefore b is set to 0.55 m. The procedure to determine the exact value of b can be further improved, but the experiments described in Section IV show that the ellipsoidal alert model is valid for positioning and tracking purposes.

B. Position estimation

Since in this paper we consider 2D-environments, only the upper half ($c \in [0, 1]$) of the ellipsoids is taken into consideration. Furthermore, every ellipsoid is weighted by the corresponding received alert magnitude so that $c \in [0, |Alert_{i,j}|]$. The monitored area is divided into fixed size pixels $p_{x,y}$ (0.125 x 0.125 m). The weight of each pixel is calculated as:

$$p_{x,y} = \sum_{N} z_{x,y} \,, \tag{6}$$

where N is the number of ellipsoids overlapping pixel $p_{x,y}$, and $z_{x,y}$ is the value of the ellipsoids at coordinates (x,y).

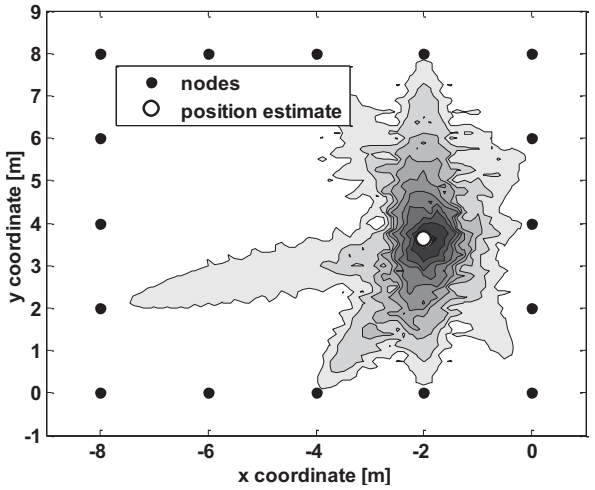


Figure 6. Estimating the position of the person by utilizing the ellipsoidal alert model.

The coordinates of the pixel having the highest weight represents the person's position estimate as illustrated in Fig. 6.

C. Kalman Filter

The position estimates contain uncertainty and noise that are caused by the noisy measurements. Therefore, the Kalman filter is an attractive solution for estimating the state of a system corrupted by noise, since under the linear Gaussian assumption it yields the minimum mean squared error (MMSE) of the system's state, i.e. the conditional mean [24]. In addition, the Kalman filter is recursive and can be updated at each new measurement, i.e. previous measurements do not have to be saved for batch processing.

When the position, velocity and acceleration of a moving object are known at a given time instant k, the position of the object at the next time instant k+1 can be estimated as follows:

$$\hat{s}_{x,k+1} = \hat{s}_{x,k} + \hat{v}_{x,k}\Delta t + \frac{1}{2}\hat{a}_{x,k}\Delta t^2 \tag{7}$$

$$\hat{s}_{y,k+1} = \hat{s}_{y,k} + \hat{v}_{y,k}\Delta t + \frac{1}{2}\hat{a}_{y,k}\Delta t^2 \,, \tag{8}$$

where Δt is the sampling period and $\hat{s}_{x,k}$, $\hat{v}_{x,k}$, $\hat{a}_{x,k}$, $\hat{s}_{y,k}$, $\hat{v}_{y,k}$ and $\hat{a}_{y,k}$ are the estimated position, velocity, and acceleration of the x and y coordinate at time instant k. The state transition matrix $\mathbf{F}$ of an object moving in a 2D-plane can be derived from (7) and (8). The state-space model of the system is:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{v}_k \,, \tag{9}$$

where $\mathbf{x}_k$ is the state vector and $\mathbf{v}_k$ is a zero-mean Gaussian process noise with covariance:

$$E\left[\mathbf{v}_k \mathbf{v}_k^T\right] = \mathbf{Q}_k \,. \tag{10}$$

Initialize Kalman filter

PREDICT

1. $\bar{\mathbf{x}} = \mathbf{F}\hat{\mathbf{x}}$	*State prediction*
2. $\bar{\mathbf{z}} = \mathbf{H}\bar{\mathbf{x}}$	*Measurement prediction*
3. $\bar{\mathbf{P}} = \mathbf{F}\mathbf{P}\mathbf{F}^T + \mathbf{Q}$	*State prediction covariance*
4. $\mathbf{S} = \mathbf{R} + \mathbf{H}\bar{\mathbf{P}}\mathbf{H}^T$	*Innovation covariance*
5. $\mathbf{W} = \bar{\mathbf{P}}\mathbf{H}^T\mathbf{S}^{-1}$	*Filter gain*
6. $\mathbf{v} = \mathbf{z} - \bar{\mathbf{z}}$	*Measurement residual*

UPDATE

7. $\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{W}\mathbf{v}$	*Updated state estimate*
8. $\mathbf{P} = \bar{\mathbf{P}} - \mathbf{W}\mathbf{S}\mathbf{W}^T$	*Updated state covariance*

GO BACK TO STEP 1.

Figure 7. The flow of the Kalman filter algorithm.

The measurement equation of the system is:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k , \qquad (11)$$

where the two-element measurement vector $\mathbf{z}_k$ contains the estimated coordinates of the tracked object. **H** is the observation model and $\mathbf{w}_k$ is a zero-mean Gaussian measurement noise with covariance:

$$E\left[\mathbf{w}_k \mathbf{w}_k^T\right] = \mathbf{R}_k . \qquad (12)$$

After obtaining the state-space matrices **F** and **H**, and the associated state- and measurement noise covariances, **Q** and **R,** the Kalman filter can be applied as illustrated in Fig. 7.

IV. Experimental Validation

This section presents the experimental results obtained with the system described in the previous sections. Four different experiments were used to validate the performance of the system. The tests were conducted in the entrance hall of a building. In each test, 16 nodes were placed at regular intervals along a square perimeter. The nodes were elevated 1.02 m from the floor using podiums. The layout of the testbeds is summarized in Table I. Fig. 8 shows two of the testbeds.

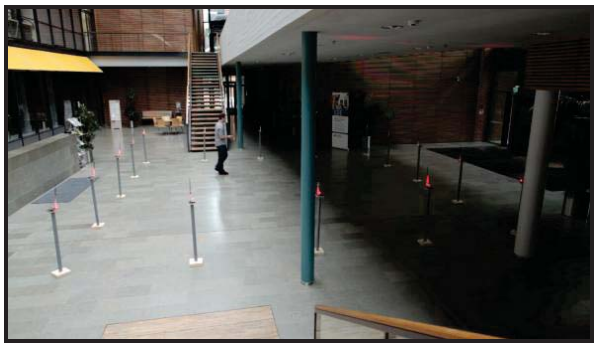
The sensor nodes used for the tests, Sensinode U100 Micro.2420 [19], utilize the IEEE 802.15.4 protocol. They run the FreeRTOS real-time kernel and the 6LoWPAN protocol stack Nano Stack v1.0.3. In order to minimize the interference of WLAN [25], the selected radio channel was 26. The transmitting power of the nodes was set to 0 dB (maximum possible value in the CC2420 radio module [26]). To prevent packets collisions and to increase the tolerance of the system to nodes failure, a TDMA communication protocol was implemented, in which each node was assigned one slot for transmission per cycle. In each packet, the transmitting node includes its unique ID, the sequence number of the packet, and the magnitude of the raised alerts (if any). The sink node listens at all times the ongoing communications; if a received packet contains alerts, these are forwarded to an external computer via UART for real-time processing.

On average, the time required for a packet to travel from the application layer of the transmitting node to the application layer of the receiving node is 7.25 ms (0.52 ms jitter). Once the content of the received packet has been transferred at the application layer, the embedded intrusion detection algorithm is executed, consuming on average 0.22 ms. In the tests, the slot length was set to 10 ms to provide enough time for the nodes to receive the packet and execute the algorithm within the dedicated time slot. In addition, an extra 10 ms slot was reserved at the end of each cycle to allow the nodes storing the raw RSSI measurements in their external flash memory. This feature enabled additional off-line analysis of the measurements. As a result, the duration of a TDMA cycle was 0.17 s.

At the beginning of a test, the monitored area was left empty. After a certain time, a person entered in it and did multiple laps along a pre-defined path which was marked on the floor. To measure the tracking accuracy of the system, the calculated position estimates are synchronized and matched to the ones extracted from a video recording. By doing this, it is possible to reconstruct the path covered by the person as a function of time.

The average error is defined as follows:

$$\varepsilon = \frac{1}{N}\sum_{k=1}^{N}\sqrt{\left(\hat{s}_{x,k} - s_{x,k}\right)^2 + \left(\hat{s}_{x,k} - s_{x,k}\right)^2} , \qquad (13)$$



(a)

(b)

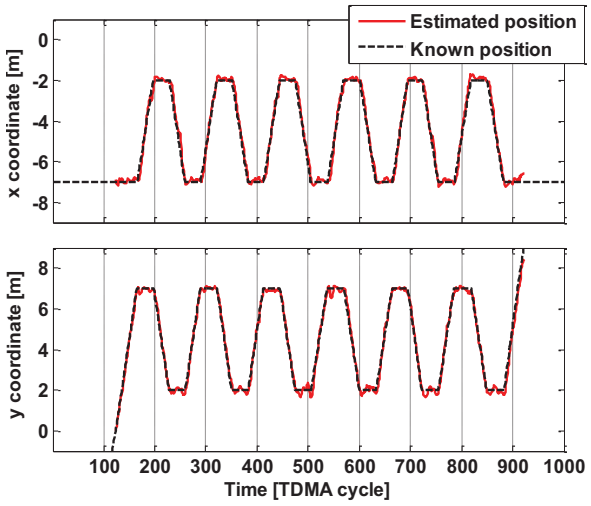
Figure 8. Test 4 (a): network monitoring a 64 m^2 area (2 m nodes interval), and test 3 (b), network monitoring a 36 m^2 area (1.5 m nodes interval).

where N is the number of position estimates, $\hat{s}_{x,k}$ and $\hat{s}_{y,k}$ are the estimated coordinates, and $s_{x,k}$ and $s_{y,k}$ are the real coordinates of the intruder at time instant k.

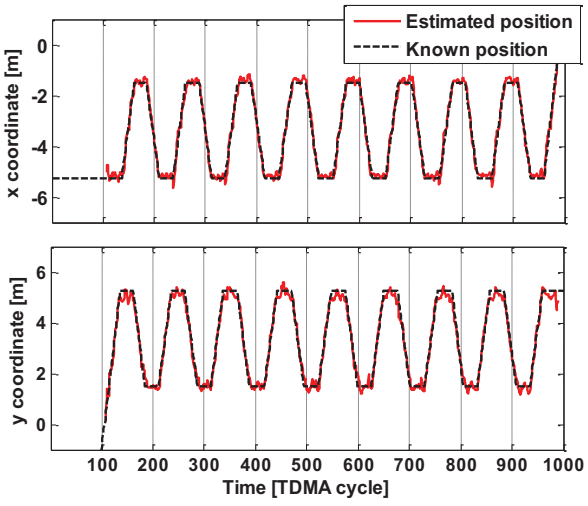
The real and estimated paths followed by the intruder in tests 3 and 4 are shown in Fig. 9. The average tracking error ε and the number of alerts transmitted by the nodes to the sink are presented in Table I. In addition, *min* ε_{lap} and *max* ε_{lap} report the average tracking error of the most and least accurate lap of the test.

The maximum tracking error is 0.24 m. Overall, the average tracking accuracy of the DFL system slightly decreases as the size of the monitored area increases, and when obstructions are found inside the monitored area. However, the difference in accuracy is in the magnitude of few centimeters.

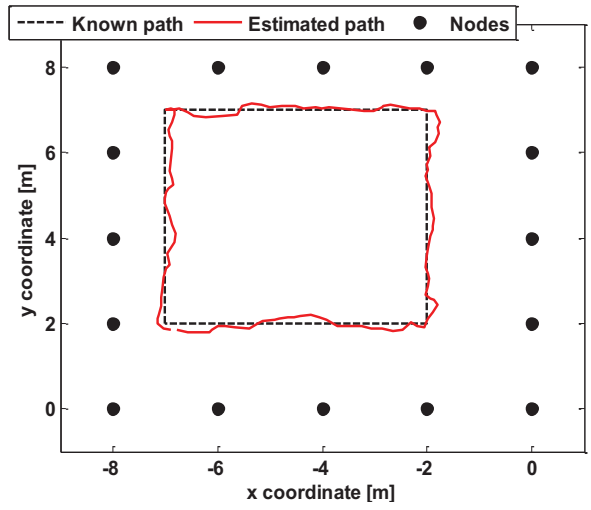
The number of alerts transmitted to the sink node is reported as a percentage of the total number of packets that would be transmitted in a system relying on centralized processing and therefore collecting the raw RSSI measurements to the sink. In Table I, $alerts_{tot}$ reports the total number of alerts transmitted during the entire test, whereas $alerts_{occ}$ indicates the number of alerts transmitted when the monitored area was actually occupied. From the results it can be observed that the number of raised alerts decreases as the size of the monitored area increases. Two facts are contributing to this phenomenon: first, with larger node intervals, the virtual RF mesh created by the wireless links becomes sparse, i.e. not as many nodes are capable of correctly detecting the person located at any given position. Second, as the nodes distance grow, the human body obstructing the links does not cause a significant attenuation and variance of the RSSI.



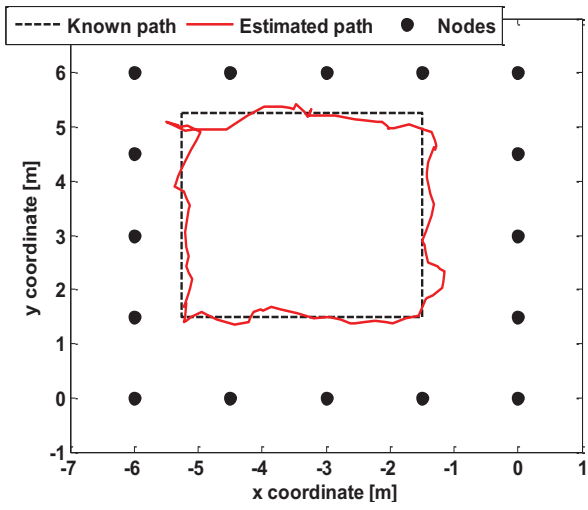
(a) Test 4, the entire path followed by the person, ε = 0.22 m.



(b) Test 3, the entire path followed by the person, ε = 0.21 m.



(c) Test 4, a single lap (TDMA cycles 260-382), ε_{cycle} = 0.23 m.



(d) Test 3, a single lap (TDMA cycles 313-416), ε_{cycle} = 0.21 m.

Figure 9. The real (dashed line) and estimated (solid line) path followed by the person during the entire tests 4 (a) and 3 (b). In (c) and (d), the real (dashed line) and estimated (solid line) path followed by the person during a single lap.

TABLE I. TESTS RESULTS

	Test 1	*Test 2*	*Test 3*	*Test 4*
Node interval [m]	1	1.5	1.5	2
Monitored area [m^2]	16	36	36	64
Obstructions	No	No	Yes	No
ε [m]	0.18	0.19	0.21	0.22
$min\ \varepsilon_{lap}$ [m]	0.15	0.18	0.20	0.19
$max\ \varepsilon_{lap}$ [m]	0.21	0.22	0.24	0.23
$alerts_{tot}$ [%]	16.4	14.6	16.4	11.7
$alerts_{occ}$ [%]	19.1	16.9	18.1	14.0

V. FUTURE WORK

So far, highest priority has been assigned to the development and implementation of the distributed RSSI processing algorithm, position estimation methods and Kalman filter. In future work, solutions to improve the tracking accuracy of the system by further improving the ellipsoidal alert model will be explored. The current research is focusing on making the system capable of correctly detecting and tracking more than a single person, also in critical three-dimensional indoor environments such as fire escapes of buildings or industrial halls with moving machineries.

An additional solution to further reduce the power consumption of the nodes is to decrease the transmission rate when no intrusions are detected. Upon the detection of an intrusion and the subsequent transmission of an alert, the transmission rate of the nodes composing the network can be once again increased. In this way, the system will be able to dynamically adjust to the ongoing situation inside the monitored area.

In a heavily obstructed environment, such as the interior of a building or an industrial hall, the attenuation of the radio signal is usually strong. This decreases the transmitting range of the nodes and the packets delivery ratio. This issue requires the design of multi-hop clustered networks, in which different clusters work simultaneously on adjacent radio channels in order to cooperatively monitor very large and obstructed areas. A reliable and real-time transmission of the alerts raised by the nodes to a central base station will need the design of an advanced networking protocol capable of forwarding large amounts of data with minimal latency.

VI. CONCLUSIONS

This work presents a WSN in which the nodes, deployed in indoor environment, detect the intrusion of a person inside the monitored area by distributed processing of the RSSI measurements. The power consumption of the system is reduced by a) minimizing the number of packets transmitted to the sink node, and b) by exploiting a high accuracy time synchronization protocol for establishing a TDMA communication protocol which enables the nodes to turn off their radio when transmissions are not scheduled. In this way, the life time of the system is at least 80 % longer than in the case of an unsynchronized network.

The system is able to accurately track in real-time the movements of a person inside the monitored area. In tests carried out with a network composed of 16 nodes covering areas of different sizes, the system was capable of tracking in real-time an intruder with an average error of 0.22 m. The number of packets transmitted to the sink node was 16.4 % of the amount that would have been transmitted in a system relying on centralized processing.

ACKNOWLEDGMENTS

This work was funded by the Finnish Funding Agency for Technology and Innovation (TEKES), in the context of the WISMII (Wireless Sensor Systems in Indoor Situation Modeling II) project [27].

The authors wish to thank Olli Viitala of the Electronic Circuit Design group of Aalto University School of Electrical Engineering for his extensive help in carrying out the time-related measurements, and Tómas Marshall for recording and editing the videos.

A video showing the performance and functioning of the system during a test can be watched at [28].

REFERENCES

[1] N. Patwari and J. Wilson, "RF Sensor Networks for Device-Free Localization: Measurements, Models, and Algorithms", in Proceedings of the IEEE, vol. 98, No. 11, Nov 2010, pp. 1961-1973, doi: 10.1109/JPROC.2010.2052010.

[2] M. Srbinovska, C. Gavrovski, and V. Dimvec, "Localization Estimation System Using Measurement of RSSI Based on Zigbee Standard," in Conference Proceedings of the 17th International Scientific and Applied Science Conference (Electronics 2008), Sep. 2008.

[3] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, "Indoor Localization System Using RSSI Measurement of Wireless Sensor Network Based on Zigbee Standard," in Conference Proceedings of the 6th International Multi-Conference on Wireless and Optical Communications, Jul. 2006.

[4] A. Faheem, R. Virrankoski, and M. Elmusrati, "Improving RSSI Based Distance Estimation for 802.15.4 Wireless Sensor Networks," in Conference Proceedings of the 2010 IEEE International Conference on Wireless Technology and Systems (ICWIT 2010), Aug. 2010.

[5] K. Benkic, M. Malajner, P. Planinsic, and Z. Cucej, "Using RSSI Value for Distance Estimation in Wireless Sensor Networks Based on ZigBee," in Proceedings of the 15th International Conference on Systems, Signals and Image Processing 2008 (IWSSIP 2008), Jun. 2008.

[6] L. Tang, K. C. Wang, Y. Huang, and F. Gu, "Channel Characterization and Link Quality Assessment of IEEE 802.15.4-Compliant Radio for Factory Environments," in IEEE Transactions on Industrial Informatics, vol. 3, No. 2, May 2007, pp. 99-110, doi: 10.1109/TII.2007.898414.

[7] K. Srinivasan and P. Levis, "RSSI is Under Appreciated", in Proceedings of the 3rd Workshop on Embedded Networked Sensors (EmNets'06), May 2006.

[8] J. Wilson and N. Patwari, "Radio Tomographic Imaging with Wireless Networks," in IEEE Transactions on Mobile Computing, vol. 9, no. 5, May 2010, pp. 621-632, doi: 10.1109/TMC.2009.174.

[9] J. Wilson and N. Patwari, "See Through Walls: Motion Tracking Using Variance-Based Radio Tomography Networks," in IEEE Transactions on Mobile Computing, to be published, Sept. 2010.

[10] D. Zhang, J. Ma, Q. Chen, and L. M. Ni, "An RF-based System for Tracking Transceiver-free Objects," in Fifth IEEE International Conference on Pervasive Computing and Communications (PerCom'07), Mar. 2007.

[11] S. Hussain, R. Peters, and D. L. Silver, “Using Received Signal Strength Variation for Surveillance in Residential Areas”, in Proceedings of SPIE Volume: 6973, Mar. 2008.

[12] K. Woyach, D. Puccinelli, and M. Haenggi, “Sensorless Sensing in Wireless Networks: Implementation and Measurements,” in Second International Workshop on Wireless Network Measurement (WinMee 2006), Apr. 2006.

[13] D. Zhang and L. M. Ni, “Dynamic Clustering for Tracking Multiple Transceiver-free Objects,” in 2009 IEEE International Conference on Pervasive Computing and Communications (PerCom’09), Mar. 2009.

[14] M. Petrova, J. Riihijärvi, P. Mähönen, and S. Labella, ”Performance Study of IEEE 802.15.4 Using Measurements and Simulations,” in 2006 IEEE Wireless Communications and Networking Conference (WCNC 2006), Apr. 2006.

[15] D. Lymberopoulos, Q. Lindsey, and A. Savvides, “An Empirical Characterization of Radio Signal Strength Variability in 3-D IEEE 802.15.4 Networks Using Monopole Antennas,” in Proceedings of the 3rd European Workshop on Wireless Sensor Networks (EWSN’2006), Jan. 2006.

[16] L. Chenyang, B.M. Blum, T.F. Abdelzaher, J.A. Stankovic, and H. Tian, “RAP: a Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks,” in Proceedings of the 8th IEEE Real-Time and Embedded Technology and Applications Symposium, Sept. 2002.

[17] H. Tian, J.A. Stankovic, L. Chenyang, and T.F. Abdelzaher, “SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks,” in Proceedings of the 23rd International Conference on Distributed Computing Systems, May 2003.

[18] O. Kaltiokallio, M. Bocca, and L. M. Eriksson, “Distributed RSSI Processing for Intrusion Detection in Indoor Environments,” in Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN ’10), poster session, Apr. 2010

[19] Sensinode, “U100 Micro.2420 Sensor Node Platform.” http://www.sensinode.com, Apr 20.2011

[20] A. Mahmood and R. Jäntti, “Time Synchronization Accuracy in Real-Time Sensor Networks,” in Proceedings of the 9th IEEE Malaysia International Conference on Communications (MICC 2009), Dec. 2009.

[21] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, ”The Flooding Time Synchronization Protocol,” in Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys ’04), Nov. 2004.

[22] CC2431 datasheet, “System-on-Chip for GHz Zigbee / IEEE 802.15.4 with Location Engine,” http://focus.ti.com/lit/ds/symlink/cc2431.pdf

[23] M. Bocca, A. Mahmood, L.M. Eriksson, J. Kullaa, and R. Jäntti, ”A Synchronized Wireless Sensor Network for Experimental Modal Analysis in Structural Health Monitoring,” in Computer-Aided Civil and Infrastructure Engineering, in press.

[24] Y. Bar-Shalom, X. Rong Li & T. Kirubarajan, "Estimation with Applications to Tracking and Navigation," John Wiley & Sons, 2001.

[25] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, “Understanding the Causes of Packet Delivery Success and Failure in Dense Wireless Sensor Networks,” in Proceedings of the 4th International Conference on Embedded Networked Sensor Systams (SenSys ’06), Nov. 2006.

[26] CC2420 transceiver datasheet, “2.4 GHz IEEE 802.15.4 / Zigbee-ready RF Transceiver.” http://focus.ti.com/docs/prod/folders/print/cc2420.html

[27] Wireless Sensor Systems in Indoor Situation Modeling http://teg.uwasa.fi/projects/wism/

[28] Wireless Sensor Systems Group http://wsn.tkk.fi/en/research/indoor_situation_awareness